

AMENDMENTS TO THE CLAIMS

OCT 17 2007

1. (Currently Amended) A method of inter-process communication between at least two application processes on a single processor of a single one computer, comprising the steps of:

5 a first process of a first application determining a name of a first file in a file system of the computer, the name of the first file being associated with a second application, the first file containing information for the first process to connect to a second process of the second application for inter-process communication;

10 the first process initiating a first connection to the second process using the information contained in the first file;

the first process communicating with the second process using the first connection if the first connection is successfully established; and

15 the first process starting a third process of the second application if the first process fails to establish a connection with the second process;

wherein said inter-process communication between at least two application processes is restricted to communications between applications running on a single processor of said a single computer under control of a single instance of an operating system.

20

2. (Original) The method of Claim 1, further comprising:

the first process initiating a second connection to the third process using the information in the first file, in response to the third process informing the first process that the third process is ready for a connection.

25

3. (Original) The method of Claim 2, wherein the third process is started in a server mode without a user interface.

4. (Original) The method of Claim 1, wherein the first process fails to establish a connection with the second process because the second process is not running.

5 5. (Original) The method of Claim 1, wherein the first file being missing from the file system indicates that the second process is not running.

6. (Original) The method of Claim 1, further comprising:

10 when the first process is started, the first process determining if a fourth process of the first application is running;

the first process requesting the fourth process to perform a task for the first process if the fourth process is running; and

the first process exiting after requesting the fourth process to perform the task for the first process.

15

7. (Original) The method of Claim 6, wherein the first process determining if the fourth process of the first application is running comprises:

20 the first process of the first application determining a name of a second file in the file system of the computer, the name of the second file being associated with the first application.

8. (Original) The method of Claim 7, wherein the second file being missing from the file system indicates that the fourth process of the first application is not running.

25

9. (Original) The method of Claim 8, wherein the second file contains information for the first process to connect to a fourth process for inter-process communication;

failure in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is not running; and

success in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is running.

10. (Original) The method of Claim 1, wherein the first process communicates with the second process using the first connection through an Application Program Interface (API).

11. (Original) The method of Claim 10, wherein the Application Program Interface (API) is platform independent.

12. (Original) The method of Claim 1, further comprising:

when the second process is started, the second process determining if a fourth process of the second application is running;

the second process requesting the fourth process to perform a task for the second process if the fourth process is running; and

the second process exiting after requesting the fourth process to perform the task for the second process.

13. (Currently Amended) An apparatus for inter-process communication between at least two application processes on a single processor of a single one computer, comprising the steps of:

a module for a first process of a first application determining a name of a first file in a file system of the computer, the name of the first file being associated with a second application, the first file containing information for the first process to connect to a second process of the second application for inter-process communication;

a module for the first process initiating a first connection to the second process using the information contained in the first file;

a module for the first process initiating communication communicating with the second process using the first connection if the first connection is successfully established;

communicating a task of said first application to said second application;

after commanding said second application to perform said task, closing said first application,

wherein multiple instances of an application program combine into one running instance of said application program; and

~~a module for the first process starting a third process of the second application if the first process fails to establish a connection with the second process.~~

wherein said inter-process communication between at least two application processes is restricted to communications between applications running on said single processor of said a single computer under control of a single instance of an operating system.

14. (Currently Amended) The apparatus of Claim 4943, further comprising:

a module for the first process initiating a second connection to the third process using the information in the first file, in response to the third process informing the first process that the third process is ready for a connection.

15. (Currently Amended) The apparatus of Claim 4943, wherein the third process is started in a server mode without a user interface.

16. (Original) The apparatus of Claim 13, wherein the first process fails to establish a connection with the second process because the second process is not running.

17. (Original) The apparatus of Claim 13, wherein the first file being missing from the file system indicates that the second process is not running.

5 18. (Original) The apparatus of Claim 13, further comprising:

a module for the first process determining if a fourth process of the first application is running, when, the first process is started;

a module for the first process requesting the fourth process to perform a task for the first process if the fourth process is running; and

10 a module for the first process exiting after requesting the fourth process to perform the task for the first process.

19. (Original) The apparatus of Claim 18, wherein the module for the first process determining if the fourth process of the first application is running comprises:

15 a module for the first process of the first application determining a name of a second file in the file system of the computer, the name of the second file being associated with the first application.

20 20. (Original) The apparatus of Claim 19, wherein the second file being missing from the file system indicates that the fourth process of the first application is not running.

25 21. (Original) The apparatus of Claim 20, wherein the second file contains information for the first process to connect to a fourth process for inter-process communication; failure in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is not running; and success in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is running.

22. (Original) The apparatus of Claim 13, wherein the module for the first process communicates with the module for the second process using the first connection through an Application Program Interface (API).

5

23. (Original) The apparatus of Claim 22, wherein the module for the Application Program Interface (API) is platform independent.

24. (Original) The apparatus of Claim 13, further comprising:

10

a module for the second process determining if a fourth process of the second application is running, when the second process is started;

a module for the second process requesting the fourth process to perform a task for the second process if the fourth process is running; and

15

a module for the second process exiting after requesting the fourth process to perform the task for the second process.

25. (Currently Amended) A program storage medium readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method of inter-process communication between at least two application processes on a single processor of a single ~~one~~ computer, the method comprising the steps of:

20

a first process of a first application determining a name of a first file in a file system of the computer, the name of the first file being associated with a second application, the first file containing information for the first process to connect to a second process of the second application for inter-process communication; and

25

the first process initiating a first connection to the second process using the information contained in the first file;

the first process communicating with the second process using the first connection if the first connection is successfully established; and

30

the first process starting a third process of the second application if the first process fails to establish a connection with the second process.

5 wherein said step of communicating communicates a task of said second application to said first application, wherein said task comprises transmitting a set of keys and values between said second application and said first application, wherein said step of communicating uses strings to transmit said keys and said values.

10 wherein said step of communicating said task of said second application to said first application further comprises use of a channel.

10 wherein said inter-process communication between at least two application processes is restricted to communications between applications running on a said single processor of said single computer under control of a single instance of an operating system.

15 26. (Original) The medium of Claim 25, wherein the method further comprises the step of:

the first process initiating a second connection to the third process using the information in the first file, in response to the third process informing the first process that the third process is ready for a connection.

20 27. (Original) The medium of Claim 26, wherein the third process is started in a server mode without a user interface.

25 28. (Original) The medium of Claim 25, wherein the first process fails to establish a connection with the second process because the second process is not running.

30 29. (Original) The medium of Claim 25, wherein the first file being missing from the file system indicates that the second process is not running.

30. (Original) The medium of Claim 25, wherein the method further comprises the steps of:

when the first process is started, the first process determining if a fourth process of the first application is running;

5 the first process requesting the fourth process to perform a task for the first process if the fourth process is running; and

the first process exiting after requesting the fourth process to perform the task for the first process.

- 10 31. (Original) The medium of Claim 30, wherein the first process determining if the fourth process of the first application is running comprises:

the first process of the first application determining a name of a second file in the file system of the computer, the name of the second file being associated with the first application.

15

32. (Original) The medium of Claim 31, wherein the second file being missing from the file system indicates that the fourth process of the first application is not running.

- 20 33. (Original) The medium of Claim 32, wherein the second file contains information for the first process to connect to a fourth process for inter-process communication;

failure in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is not running; and

25

success in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is running.

34. (Original) The medium of Claim 25, wherein the first process communicates with the second process using the first connection through an Application Program Interface (API).

5 35. (Original) The medium of Claim 34, wherein the Application Program Interface (API) is platform independent.

36. (Original) The medium of Claim 25, wherein the method further comprises the steps of:

10 when the second process is started, the second process determining if a fourth process of the second application is running;

the second process requesting the fourth process to perform a task for the second process if the fourth process is running; and

15 the second process exiting after requesting the fourth process to perform the task for the second process.

37. (New) The method of Claim 1, wherein said process determines a name of a rendezvous file from a filename of said first application, wherein said second application locates said rendezvous file, without prior knowledge of a running instance of said first application program, through use of said filename.

20

38. (New) A method of inter-process communication between at least two application processes on a single processor of a single computer, comprising the steps of:

25 initiating a first application program on said single processor;

upon initiating a second application program, said second application program determining if a prior instance of the second application program is running on said single processor;

if said first application program comprises a prior instance of said second application program and said first program is still running, performing all of:

establishing an inter-process communication channel between said second application program and said first application program;

5 communicating a task of said second application program to said first application program;

after commanding said first application program to perform said task, exiting said second application program,

10 wherein multiple instances of an application program combine into one running instance of said application program.

39. (New) The method of Claim 38, wherein said step of said second application program determining if a prior instance of said second application program is running on said single processor comprises use of a rendezvous file.

15

40. (New) The method of Claim 39, wherein said rendezvous file comprises a filename named after said first application program, wherein said second application program can locate said rendezvous file, without prior knowledge of a running instance of said first application program, through use of said filename.

20

41. (New) The method of Claim 40, wherein said rendezvous filename comprises placement in a predetermined location, wherein said second application program can locate said rendezvous file, without prior knowledge of a running instance of said first application program, through use of both said filename and said location.

25

42. (New) The method of Claim 39, further comprising a step of said second application process computing a filename of said rendezvous file from a name of said second application process and using said filename in said step of establishing said inter-process communication channel between said second application program and said first application program.

30

43. (New) The method of Claim 38, wherein initiating multiple instances of said application program results in one and only one instance of said application program running on said single processor of said computer.

5

44. (New) The method of Claim 38, wherein said step of communicating a task of said second application program to said first application program comprises transmitting a set of keys and values between said second application program and said first application program.

10

45. (New) The method of Claim 44, further comprising a step of using strings to transmit said keys and said values.

15

46. (New) The method of Claim 38, wherein said step of communicating said task of said second application program to said first application program comprises use of a channel.

20

47. (New) The method of Claim 46, wherein said channel connects a server process of said second application program to a client process of said first application program.

25

48. (New) The method of Claim 38, wherein said method of inter-process communication comprises communication based upon a filename, wherein a software application based upon said method is not operating system-dependent when said operating system provides access to files.

49. (New) The apparatus of Claim 13, further comprising a step of:
a module for the first process starting a third process of the second application if the first process fails to establish a connection with the second process.